

66/60/10
JC408
U.S.

**UTILITY
PATENT APPLICATION
TRANSMITTAL**

(Only for new nonprovisional applications under 37 CFR 1.53(b))

Attorney Docket No.

36J.P207

First Named Inventor or Application Identifier

ANDREW J. KRASLAVSKY

Express Mail Label No.

USPTO
09/288967
04/09/99

APPLICATION ELEMENTS

See MPEP chapter 600 concerning utility patent application contents.

1. Fee Transmittal Form
(Submit an original, and a duplicate for fee processing)
2. Specification Total Pages **28**
3. Drawing(s) (35 USC 113) Total Sheets **10**
4. Oath or Declaration Total Pages **2**

- a. Newly executed (original or copy)
- b. Unexecuted for information purposes
- c. Copy from a prior application (37 CFR 1.63(d))
(for continuation/divisional with Box 17 completed)
[Note Box 5 below]

- i. **DELETION OF INVENTOR(S)**
Signed Statement attached deleting
Inventor(s) named in the prior application,
see 37 CFR 1.63(d)(2) and 1.33(b).

5. Incorporation By Reference (*useable if Box 4c is checked*)
The entire disclosure of the prior application, from which a copy of
the oath or declaration is supplied under Box 4c, is considered as
being part of the disclosure of the accompanying application and is
hereby incorporated by reference therein.

ADDRESS TO: Assistant Commissioner for Patents
Box Patent Application
Washington, DC 20231

6. Microfiche Computer Program (Appendix)
7. Nucleotide and/or Amino Acid Sequence Submission
(if applicable, all necessary)
- a. Computer Readable Copy
 - b. Paper Copy (identical to computer copy)
 - c. Statement verifying identity of above copies

ACCOMPANYING APPLICATION PARTS

8. Assignment Papers (cover sheet & document(s))
9. 37 CFR 3.73(b) Statement
(when there is an assignee) Power of Attorney
10. English Translation Document *(if applicable)*
11. Information Disclosure Statement (IDS)/PTO-1449 Copies of IDS Citations
12. Preliminary Amendment
13. Return Receipt Postcard (MPEP 503)
(Should be specifically itemized)
14. Small Entity Statement(s) Statement filed in prior application
Status still proper and desired
15. Certified Copy of Priority Document(s)
(if foreign priority is claimed)
16. Other: _____

17. If a CONTINUING APPLICATION, check appropriate box and supply the requisite information:

Continuation Divisional Continuation-in-part (CIP) of prior application No. ___/___

18. CORRESPONDENCE ADDRESS

<input checked="" type="checkbox"/> Customer Number or Bar Code Label		05514 (insert Customer No. or Attach bar code label here)		or <input type="checkbox"/> Correspondence address below	
NAME _____					
Address _____					
City _____	State _____	Zip Code _____			
Country _____	Telephone _____	Fax _____			

CLAIMS	(1) FOR	(2) NUMBER FILED	(3) NUMBER EXTRA	(4) RATE	(5) CALCULATIONS
	TOTAL CLAIMS (37 CFR 1.16(c))	28 - 20 =	8	X \$ 18.00 =	\$ 144.00
	INDEPENDENT CLAIMS (37 CFR 1.16(b))	1 - 3 =	0	X \$ 78.00 =	\$ 0.00
	MULTIPLE DEPENDENT CLAIMS (if applicable) (37 CFR 1.16(d))			\$260.00 =	\$ 260.00
				BASIC FEE (37 CFR 1.16(a))	\$ 760.00
				Total of above Calculations =	\$1,164.00
				Reduction by 50% for filing by small entity (Note 37 CFR 1.9, 1.27, 1.28).	
				TOTAL =	\$1,164.00

19. Small entity status

- a. A Small entity statement is enclosed
- b. A small entity statement was filed in the prior nonprovisional application and such status is still proper and desired.
- c. Is no longer claimed.

20. A check in the amount of \$ 1,164.00 to cover the filing fee is enclosed.

21. A check in the amount of \$ 40.00 to cover the recordal fee is enclosed.

22. The Commissioner is hereby authorized to credit overpayments or charge the following fees to Deposit Account No. 06-1205:

- a. Fees required under 37 CFR 1.16.
- b. Fees required under 37 CFR 1.17.
- c. Fees required under 37 CFR 1.18.

SIGNATURE OF APPLICANT, ATTORNEY, OR AGENT REQUIRED

NAME	Brian L. Klock - Registration No. 36,570
------	--

SIGNATURE	
-----------	--

DATE	April 9, 1999
------	---------------

666000-79688360

5

NEGOTIATING AN EXCHANGE OF
IMAGE PROCESSING FUNCTIONALITY

BACKGROUND OF THE INVENTION

10

Field Of The Invention

The present invention relates to transmission of image data between two image processing devices over a bi-directional communication interface, such as a local area network. More specifically, the invention involves negotiation between two devices such as an image output device and an image input device, so as to transfer image data between the devices, and so as to negotiate whether exportable image processing functionality on one device should be exported to the other device so as to increase overall image transfer efficiency.

15

20

Description Of The Related Art

In local area networks and other environments providing bi-directional communication

between connected devices, it is becoming increasingly common for such devices to communicate directly with each other so as to intelligently negotiate information exchange ordinarily without the need of a separate computer providing oversight.

Once such negotiation protocol is JetSend™, which is a device-to-device communications protocol for local and wide area networks that allows network devices to intelligently negotiate information exchange.

The JetSend™ protocol allows the two devices to connect over a network, to negotiate a common data type for the information exchange, to provide device status, and to exchange the needed information, all without intervention from a user or from a third device such as a computer which provides oversight over the communication between the two devices.

JetSend™ protocol is described at length in "HP JetSend Communications Technology, Section I: Architectural Overview; Section II: Protocol References; and Section III: E-Material Specification", Version 1.0, Hewlett Packard Co., 1997. These documents are incorporated by reference as if set forth in full herein.

One common example of information exchange is the exchange of image data between an image input device (such as a scanner or digital camera) and an image output device (such as a printer or facsimile or optical projector). According to currently-accepted information exchange negotiations, such as the JetSend™ negotiation, the image input device and the image output device jointly negotiate for a common data type for the image data, based on the image processing capabilities of the input and output devices, and thereafter exchange the image data in accordance with the negotiated data type. For example, in the case of an image data exchange between a color scanner and a monochrome printer,

00000000000000000000000000000000

the scanner might include image processing capabilities so as to provide image data according to data formats that include full color image data and grayscale image data. The printer, on the other
5 hand, might include image processing capabilities that allow it to accept full color image data, grayscale image data, and black-and-white image data, and thereafter to process such data so as to printout a monochrome image. For this contemplated
10 information exchange, therefore, the most efficient data type for transfer of image data is grayscale image data, since it is commonly exportable and receivable by both the scanner and the printer, and since it results in the transmission of the least amount of data (and consequently the least amount of
15 transmission time). Accordingly, for this information exchange, the scanner would scan in an image, convert the image into grayscale data, and transmit the grayscale data over the bi-directional communication interface to the printer; whereupon the printer would receive the grayscale image data from the bi-directional communication interface,
20 convert the grayscale image data to monochrome image data, and print out the monochrome image data.
25

While such negotiations are advantageous in that they allow devices to negotiate a commonly-acceptable format for data transfer, improvements in efficiency are still needed. Specifically, because such negotiations fail to consider the relative
30 processing power between the two devices, such negotiations often lead to a data transfer that is inefficient and time consuming.

SUMMARY OF THE INVENTION

35

It is an object of the invention to address the foregoing, by considering the relative

processing capabilities between two devices involved
in an information exchange on a bi-directional
communication link, and by exchanging data
processing functionality between the devices so as
to take advantage of better performance on one
device relative to that of another.

Accordingly, in one aspect, the invention
is a negotiation for an exchange of image processing
functionality between two devices communicating over
a bi-directional communication link. The devices
exchanges device class modules which include
information concerning the relative processing power
of the two devices, and thereafter exchange function
code descriptions which include information
concerning the functionality available in each
device, together with information concerning whether
such functionality is exportable to other devices.
Thereafter, the two devices negotiate an assignment
of functionality between the two devices, so as to
effect an efficient image data transfer. As part of
the negotiation, the devices determine whether an
exchange of functionality is needed, and if such an
exchange is needed, the devices transfer the
functionality in the form of executable code. Data
transfer is thereafter effected using the negotiated
assignment of functionality between the two devices,
including functionality that has been exported from
one device to another.

In one preferred embodiment, negotiation of
assignment of functionality involves a determination
of alternative processing sequences for image data
transfer. For example, image data transfer often
involves the sequential application of individual
image processing functions, such as functions to
convert image data from one format to another, to
compress and decompress the image data, to change
the color depth (or to eliminate color), or to apply

color correction functions. Thus, to negotiate an assignment of functionality to effect image data transfer, a determination is made of these various alternatives. A cost function is applied to each alternative, with the cost function considering cost in terms of data transfer time and image quality. The processing sequence with the lowest cost is then selected, with its attendant exchange of processing functionality from one device to another.

In further preferred aspects, it is possible for each of the two devices involved in the information exchange to refer to a third device on the bi-directional communication link, with the third device including a repository of image processing functionality. In this situation, the negotiation to assign functionality to effect image data transfer can proceed by additional functionality provided by the repository of image processing functions, so as to achieve and even further increase efficiency in image data transfer.

This brief summary has been provided so that the nature of the invention may be understood quickly. A more complete understanding of the invention can be obtained by reference to the following detailed description of the preferred embodiment thereof in connection with the attached drawings.

BRIEF DESCRIPTION OF THE DRAWINGS

Figure 1 is a topological view of a representative wide area network to which the present invention may be applied.

Figure 2 is a block diagram of a representative network interface.

Figure 3 illustrates the architecture of software in the network interface.

Figure 4 illustrates the architecture of software for an image output device.

Figure 5 illustrates the content of the device class module.

5 Figure 6 illustrates the content of the function code description module.

Figure 7 is a flow diagram for explaining negotiation processing.

10 Figures 8, 9A and 9B are views for a detailed example of an exchange between a scanner and a printer.

Figure 10 is a flow diagram for explaining an alternative embodiment of the invention.

15 DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT

The invention is applicable in any environment for which a two-way communication interface is available between two remote devices. Once such environment is a local or wide area network that connects devices having JetSend™ protocol. Accordingly, the invention is described herein in connection with the JetSend™ protocol, with the understanding that any two-way communication interface is equally usable.

Figure 1 is a topological view of a representative wide area network to which the present invention may be applied. Shown in Figure 1 are three separate local area networks (LAN's) 100, 110 and 120, connected to each other through modems 130 and backbone 140 into an overall wide area network (WAN). Each of LAN's 100, 110 and 120 may be configured physically into any of the known network configurations such as Ethernet or Token Ring, and each LAN may carry network transmissions using one or more network protocols over one or more frame types. Because of the diversity of protocols

and frame types, the wide area network of Figure 1 is often referred to as a "heterogeneous" network.

For purposes of emphasis, it is further assumed that at least some of the LAN's are physically dissimilar; accordingly, in this representative embodiment, LAN's 100 and 110 are configured as Ethernet networks while LAN 120 is configured as a Token Ring network.

Connected to each of the LAN's are plural workstations and plural networked devices. For example, connected to LAN 100 are workstations 103 and 108, together with networked devices including printer 102, scanner 104, and printers 105 and 109. Also connected to LAN 100 is file server 106 which manages access to files stored on large capacity network disk 107. Other unshown networked devices and workstations may also be connected to LAN 100.

Connected to LAN 110 are workstations 111 and 112, and networked devices including printers 115 and 116 and multi-function (print, scan and fax) device 118. (Actually, multi-function device 118 is connected to LAN 110 through a network interface 119, as explained more fully below.) LAN 110 also includes file server 113 which manages access to large capacity network disk 114.

Connected to LAN 120 are workstations 121 and 122, as well as networked devices such as printer 116.

Each of the networked devices includes a network interface such as a network interface card ("NIC") or a stand-alone network interface. The function of the network interface is to provide an electrical and software interface to two-way communication on the network, specifically to receive and to send network transmissions to and from the networked device. For example, in the case of printer 102 which includes a NIC, the function of

the NIC is to receive print jobs from LAN 100, such
as a print job from a print queue in file server
106. Likewise, another function of the NIC for
5 printer 102 is to send transmissions from printer
102 to LAN 100, such as transmissions which poll for
the presence of print jobs in the print queue. Such
transmissions of the network interface will
hereinafter be referred to as "device-specific"
10 transmissions, since they relate to the purpose of
the networked device.

At least some of the devices on the WAN are
JetSend™-enabled devices. For example, scanner 104
and printer 115 might be JetSend™-enabled, meaning
that such devices can communicate directly over the
15 WAN, without the intermediary of some other
computing device such as one of the workstations, so
as to cause an original document to be scanned in by
scanner 104 and printed by printer 115. Other
JetSend™-enabled devices, both input and output, are
20 also possible, such as a JetSend™-enabled networked
camera, a JetSend™-enabled networked whiteboard, a
JetSend™-enabled networked facsimile, a JetSend™-
enabled networked projector, a JetSend™-enabled
networked workstation, and the like. JetSend™
25 communications between such devices will hereinafter
be referred to as "JetSend™" transmissions.

Each networked device on the WAN includes
some network interface. For example, printers 102,
105, 109 and 115 each include a single NIC. Scanner
30 104 includes a single NIC. Multi-function device
118 includes stand-alone network interface 119 which
includes a multi-function controller 119a. JetSend™
software operates on the network interface.

Figure 2 is a block diagram of a
35 representative network interface, such as a network
interface card, for interfacing a networked device
to a network. As shown in Figure 2, the network

interface includes central processing unit (CPU) 11 such as an Intel 80188 8-bit programmable microprocessor which is interfaced to computer bus 12. Also interfaced to computer bus 12 are non-volatile RAM (NVRAM) 14, dynamic RAM (DRAM) 15, electrically erasable programmable read only memory (EEPROM) 16, drivers for LED indicators 17, and serial interface 19 which includes both a transmit and a receive terminal. Configuration parameters and other important information concerning the configuration of the network interface are ordinarily stored in NVRAM 14, where such information may be stored between power cycles.

Process steps for execution by microprocessor 11 are stored in EEPROM 16, but in general, before those process steps are executed, they are copied from EEPROM 16 into DRAM 15 for execution out of DRAM 15. LED indicators 17 are provided so as to give a user a general indication as to the processing status of the network interface, and may include, for example, indicator LEDs for power and data communication. Serial interface 19 is provided so as to allow local serial access to the network interface.

As mentioned previously, the interface is between the local area network and the networked device, and the network interface is therefore usually provided with a connector 20 for connecting to the LAN as well as a connector 21 for connecting to the networked device. On the network side, a network interface controller 22 is connected to bus 12 and to network connector 20, and provides hardware interface for capturing and assembling network transmissions received by the network interface and transmitted onto the LAN. Static RAM (SRAM) 24 is provided for network interface controller 22 so as to store, on a temporary basis, such network transmissions.

On the peripheral side, peripheral
interface controller 25, which is connected to bus
12 and peripheral connector 21, provides peripheral
interface between the network interface and the
5 networked device serviced by the network interface.
Peripheral controller 25 may be of any of a variety
of different controllers, and can, for example, be
constructed for a bi-directional interface (such as
10 a SCSI interface, an IEEE 1284 interface, a dual
port RAM interface, a shared RAM interface, and the
like) or a uni-directional interface (such as a
standard Centronix RS-232 interface or the like).
Peripheral interface controller 25 is provided with
15 SRAM 26 for temporary storage of data transmitted
between the network interface and the networked
device.

Software to implement a negotiation of an
exchange of image processing functionality,
according to the invention, is stored on and
operates on the network interface card shown in
20 Figure 2, as described more fully below in
connection with Figure 3.

Figure 3 illustrates the architecture of
software for an image input device such as scanner
25 104, as stored in and executed by the NIC. Such
software is stored as computer-executable process
steps in a computer readable medium, such as the
aforementioned EEPROM 16. As shown in Figure 3, the
30 architecture of the software extends through the NIC
between the LAN and the networked input device, so
as to provide interface for the networked device to
the LAN. Thus, the architecture of the software
includes physical layer 31, a plurality of different
protocol stacks 32, a negotiation controller 33
35 according to the invention, JetSend™ interaction
protocol 34, and device-specific applications such
as scanner control application 35. Device-specific

applications are applications concerning the functionality of the network device, such as the aforementioned scanner control application 35.

5 JetSend™ interaction protocol 34 provides for communication between JetSend™-enabled devices, and preferably is implemented using a number of layered protocols, with each protocol layer communicating with the corresponding layer in the other device. The layers preferably include
10 interaction policies, interaction protocol, session protocol and reliable messaging protocol.

Plural protocol stacks 32 are preferred since they increase the flexibility of a networked device by allowing it to communicate via the network interface on the LAN using plural different network protocols. In the Figure 3 example, two different network protocol stacks are shown, including IPX protocol stack 36 and IP protocol stack 37. Other protocol stacks may also be provided, such as a DDP protocol stack, a UDP protocol stack, a NetBIOS protocol stack, or an AppleTalk protocol stack. Of course, it is possible for a networked device to operate over even a single protocol stack.
20

Figure 4 illustrates the architecture of software for an image output device such as printer 115, as stored in and executed by the NIC for the printer. In much the same way as the software for image input devices, such software is stored as computer-executable process steps in a computer readable medium, such as in EPROM 16. As shown in Figure 4, the architecture of the software extends through the NIC between the LAN and the networked output device, so as to provide interface for the networked device to the LAN. Thus, the architecture of the software in Figure 4 includes physical layer 41, a plurality of different protocol stacks 42, device-specific applications such as printing
25
30
35

control applications 44, a JetSend™ interface protocol 45, and a negotiation controller 46 according to the invention. The NIC further includes a printer interface module such as XP module 47. XP module 47 provides process steps for a standard interface between the NIC and the printer. In particular, and as illustrated in Figure 4, the printer includes a print engine and a print controller, and XP module 47 interfaces directly to the print controller over whatever interface is provided therebetween, such as the aforementioned bi-directional interfaces or the aforementioned uni-directional interfaces.

In much the same way as plural protocol stacks increase the flexibility of a networked input device, plural protocol stacks 42 increase the flexibility of a networked output device such as the printer, by allowing the NIC to communicate via the network interface using plural different output protocols. Again, in Figure 4, two different protocol stacks are shown including IP protocol stack 48 and IPX protocol stack 49. Other protocol stacks may also be provided; and it is further possible for the networked output device to operate over even a single protocol stack.

In the Figure 4 example, printer control application 44, JetSend™ interaction protocol 45 and negotiation controller 46 are all shown as operating over the IP protocol stack 48. It is to be understood that this is representative only, and it is typical for printer control applications, JetSend™ interface protocols, and negotiation controllers also to operate over each of the different protocol stacks included in the NIC.

JetSend™ interaction protocol 45 is similar to but complementary to JetSend™ interaction protocol 34.

Each of negotiation controllers 33 and 46 respectively stores two types of data modules that are exchanged during negotiation processing between two devices so as to determine whether or not image processing functionality of one device is transmitted to the other device for execution by that other device. The two data modules are a device class module and a function code description code. Ordinarily, only a single device class module is included for each different device; however, for function code description modules, many function code description modules are included in each device, corresponding to each of the different functions that the device performs. Figures 5 and 6 illustrate the content of the device class module and the function code description module.

As shown in Figure 5, the device class module stores information concerning processing power of the device, as well as information identifying processor type. Thus, as shown in Figure 5, device class module 50 includes fields respectively storing device type 51 (such as "scanner" or "printer"), device processing power 52 (such as a numeric value indicating relative processing capability of the device), platform CPU 54 (such as the specific manufacturer of microprocessor 11), platform operating system (such as various real-time operating systems like VxWorks or pSOS), and storage availability 56 (such as the total amount of storage available in DRAM 15 beyond that which is needed for default processing of the device).

As shown in Figure 6, the function code description module includes information pertaining to each of the different functions that the device is capable of performing, as well as information concerning whether such functions are exportable to

other devices together with information that can be used to determine whether exportability is desired. Thus, as shown in Figure 6, function code
5 description module 60 includes fields for a function identification 61 (such as compression or decompression capability, image conversion capability, color correction capability, and the like), version identifier 62 so as to identify the specific version of the function in question, a name field 64 which gives an alphanumeric reading of the function in question, a purpose field 65 which gives an alphanumeric explanation of the purpose of the function in question, a processing cost factor 66 (such as a numeric factor that indicates the relative processing complexity of the processing in question), a size field 67 which gives the amount of storage needed to execute the function in question, and an exportability field 68 which gives an indication as to whether or not the function in
10 question is exportable to another device, as well as information as to whether or not the function in question is platform independent or platform dependent.
15
20

Figure 7 is a flow diagram for explaining
25 negotiation processing according to the invention, and specifically negotiation processing to determine whether or not image processing functionality should be exported from one device for execution on another device in connection with a transfer of image data between the two devices. The process steps shown in Figure 7 are computer-executable process steps
30 preferably stored on a computer readable medium, such as EEPROM 16 or DRAM 15. Briefly, according to Figure 7, the process steps operate to exchange device information and exchange exportable function
35 information between the two devices, to negotiate an assignment of functions between the two devices so

as to effect a transfer of image data, to exchange exportable functions as needed, and to transfer image data according to the negotiated assignment of functions.

5 In more detail, in step S701, the first device and the second device connect over the bi-directional communication link and establish two-way communication with each other. Preferably, the devices are an image input device, such as a scanner or a digital camera, and an image output device, such as a printer or facsimile or overhead projector.

10 In step S702, the first and second devices exchange device class modules, thereby providing the other device with information concerning device capabilities and platform.

15 In step S704, the first and second devices exchange function code descriptions, thereby providing the other device with information concerning the available image processing functions as well as whether or not such image processing functions are exportable. Thus, the first device provides the second device with a function code description for each of its image processing functions including, for example, image processing functions such as color conversion functions, compression and decompression functions, color manipulation functions, and image conversion functions. The second device likewise provides the first device with a function code description for each of its image processing functions.

20 In steps S705 through S707, the first and second devices negotiate an assignment of functionality so as to effect the transfer of image data. Specifically, in steps S705 through S707, the first and second devices mutually determine various alternatives for image processing functionality

needed so as to transfer image data from the first device to the second device. This determination is made without regard to where specific image processing functionality might reside, and thus contemplates the overall image processing functionality needed to effect the data transfer. For each of the different alternatives, the devices apply a cost function so as to determine which alternative is the lowest cost. For example, the cost function might involve a simple determination of the quickest time to transfer the image data, or the cost function might involve a determination of which alternative provides the best image quality. It is also possible to provide a cost function which contemplates a tradeoff between image transfer time and image quality, such that a reasonably good image is transferred in the least amount of time.

In more detail, in step S705, the devices determine the various alternative processing sequences that are needed to effect a transfer of image data. For example, to transfer image data from a color scanner to a monochrome printer, one possible alternative to a processing sequence is to convert color data directly to monochrome data, to transfer the monochrome data to the printer, and to print the monochrome data at the printer. A different alternative processing sequence is to convert the color data to grayscale data to compress the grayscale data, to transmit the grayscale data to the printer, to decompress the grayscale data, to convert the decompressed grayscale data to monochrome data, and to print the monochrome data. Each of these different alternative processing sequences for a transfer of image data is developed by the devices in this step S705.

In step S706, a cost function is applied to each of the different alternatives that were

determined in step S705. A generalized cost function, which contemplates a tradeoff between reasonably good image quality coupled with reasonably quick transfer time, is as follows:

5 Cost Function = (Time to transfer image processing functionality between devices) + (Time to execute image processing functionality in each device) + (Time to transfer data) + (Image quality factor).

10 The foregoing cost function is applied to each of the different alternatives determined in step S705.

In step S707, the image processing sequence with the lowest cost function is selected.

15 It should be noted that although these steps S705 through S707 ordinarily contemplate a determination of various alternative processing sequences by both the first and second devices, it is possible for these steps to be performed only in one device. In such a situation, it is ordinarily preferable for the processing steps to be performed in the sending device, for the reason that the sending device ordinarily has more information about the desires of the user in effecting the data transfer.

20 Step S709 determines whether an exchange of functionality is needed, based on the selected image processing sequence from step S707. Specifically, if the selected image processing sequence contemplates an exchange of image processing functionality from one device to another, then flow branches to step S710 to exchange the needed functionality. In any event, flow then advances to step S711 in which the image data is processed according to the selected processing sequence and transferred from the first device to the second device. In step S712, the devices disconnect.

5 Figures 8, 9A and 9B are views for a detailed example of an exchange between a scanner and a printer. Thus, as shown in Figure 8, following connection between the scanner and the printer (step S701), the scanner and the printer exchange device class modules (step S702). As shown in Figure 8, the scanner sends its device class module indicating that it is a scanner, with a processing power factor of 12, a StrongARM 10 processing unit, and VxWorks operating system. 400 Kb of memory is available in the scanner for processing capabilities beyond those ordinarily needed by the scanner itself.

15 Likewise, the device class module from the printer indicates that it is a monochrome laser printer with a processing power factor of 8, a MIPS microprocessor and a pSOS operating system. 256 Kb of storage is available on the printer for processing beyond the normal needs of the printer 20 itself.

25 After exchanging device class modules, the printer and the scanner exchange function code descriptions. Function code descriptions for the scanner are illustrated in Figure 9A, and indicate that the scanner has three functions: image conversion from 24-bit color to 8-bit grayscale; image conversion from 8-bit grayscale to black-and-white; and JPEG decompression. For image conversion from 24-bit color to 8-bit grayscale, the function 30 code description indicates that the processing cost factor is 15, the size needed is 64 Kb, the function is exportable, and the function is platform dependent (meaning that it can execute only on a platform and operating system that is the same as that of the scanner which is VxWorks on a StrongARM CPU). Likewise, for image conversion for 8-bit 35 grayscale to black-and-white, the function code

description indicates that the processing cost factor is 12, the size is 85 Kb, the function is exportable, and it is platform independent (meaning that it can be executed on any operating system or any CPU). Platform independent code is ordinarily written in a platform independent language such as JAVA. Finally, for JPEG decompression, the function code description indicates that its processing cost factor is 20, its size is 90 Kb, and that it is not exportable.

Figure 9B illustrates the function code descriptions provided by the printer to the scanner. As shown in Figure 9B, the printer reports two functions: image conversion from 8-bit grayscale to black-and-white; and JPEG-LS compression and decompression. For image conversion from grayscale to black-and-white, the function code description indicates that its processing cost is 10, its size is 40 Kb, and it is not exportable. For JPEG-LS compression and decompression, the function code description indicates that its processing cost is 25, its size is 100 Kb, it is exportable and platform independent.

Following exchange of function code descriptions, the printer and the scanner then commence a negotiation of assignment of functionality so as to effect transfer of image data from the scanner to the printer (steps S705 through S707). In this case, the printer and the scanner (or the scanner alone, in the case where only one device makes the final selection of a processing sequence) immediately determine that JPEG-LS is the optimal compression choice for compression, since JPEG-LS is a lossless scheme employing a higher image quality than JPEG alone. Moreover, although its processing cost is high when executed in the printer (as indicated by the processing cost of 25

seconds at the printer power rating of 8), cost
function analysis described below will indicate that
its processing cost can be reduced to $(25 \times 8)/12 =$
16.66 if JPEG-LS is executed in the scanner, because
of the scanner's higher processing power (here, 12).
In addition, the scanner has sufficient available
storage space (400 Kb) to house the JPEG-LS
function.

In addition, in determining alternatives
for processing sequences for image data transfer,
since the printer is not capable of printing color,
the image must be converted from 24-bit color to
black-and-white data. However, the conversion from
24-bit color to black-and-white data need not be
done in a single step, because the printer has image
conversion capability from a 8-bit grayscale to
black-and-white. Accordingly, it is possible for
the scanner to convert from 24-bit color to 8-bit
grayscale, and for the printer to convert from 8-bit
grayscale to black-and-white.

Moreover, since the scanner is the more
powerful device and already owns a function code to
convert color from 24-bit color to 8-bit grayscale,
there is no need to transfer the function code to
the printer to perform such a conversion from 24-bit
color to 8-bit grayscale. However, if the printer
had been the more powerful device, negotiation may
have resulted in a determination that the color
conversion function codes should be transferred to
the printer so that its superior processing power
would increase the overall speed of image data
transfer. This, of course, would have been weighed
against the cost of transferring the original image
as 24-bit color (which obviously is significantly
higher than transferring the image as 8-bit
grayscale data).

Conversely, if the scanner had not possessed a function code to conduct color conversion, but the printer did, a comparison similar to the above would most likely have resulted in a determination that the function code for color conversion should be transferred from the printer to the scanner, for execution there.

It is further noted that although both the printer and the scanner have function codes for conversion of grayscale images to black-and-white, the printer's version is older (version 3.00.00 as compared to version 2.00.23). If the image processing requirements for the current job would require it, and if such a transfer were included in the cost function, then transfer of the scanner's version to the printer would have been effected.

Overall, the cost function applied to this particular negotiation, and applied to each of the different processing alternatives, is as follows:

Cost function = (Time to transfer function and to install it) + (Time to transfer image data) + (Time to execute function codes at each of the printer and scanner).

In this particular example, and as indicated above, it is determined that the lowest cost processing alternative is an alternative where the scanner converts 24-bit color to grayscale data, compresses the grayscale data according to JPEG-LS compression that it obtains from the printer, and transfers the compressed grayscale data to the printer; whereafter the printer decompresses the compressed grayscale data, converts the decompressed grayscale data to black-and-white image data, and prints out the image. Accordingly, and in accordance with steps S709 and S710, the printer transfers its functionality for JPEG-LS compression to the scanner. Thereafter, the data transfer job

is processed according to the agreed-upon image processing sequence, resulting in a scan at the scanner and a printout at the printer.

5 Figure 10 is a flow diagram for a second embodiment of the invention. One way that the second embodiment differs from the first embodiment is that functionality need not be obtained only from the devices involved in the transfer of image data. Rather, functionality can also be obtained from a
10 repository of functionality elsewhere on the network, such as from a file server or a computer workstation, or from another image processing device on the network that is not involved in the transfer of image data in question from the first device to the second device.
15

20 Thus, for example, and referring briefly to Figure 1, in a situation where a transfer of image data is contemplated between scanner 104 and printer 116, it is possible to obtain image processing functionality from a repository on a third device, such as file server 106, workstation 121, or multifunction device 118.

25 Referring again to Figure 10, in step S1001, the first and second devices connect and in steps S1002 and S1003 the devices exchange device class modules and function code descriptions. In step S1004, the first and second devices obtain function code descriptions from a repository or repositories of image processing functionality.
30 Thereafter, in steps S1005 through S1007, the devices negotiate an assignment of functionality to effect the transfer of image data, this time considering both functionality present in the first and second devices, together with functionality present in the repository or repositories. Thus, in step S1005, the devices determine alternative processing sequences for the transfer of image data,
35

666070-29688360

in step S1006, the devices apply a cost function to each different alternative, and in step S1007, the devices select the processing alternative with the lowest cost function.

5 In step S1009, the devices determine whether functionality not currently present in the first and second devices is needed in order to process the image data according to the selected alternative. If functionality is needed, then flow 10 branches to step S1010 in which the functionality is obtained from the other device, or from the repository of image processing functionality, as appropriate. Thereafter, flow branches to step 15 S1011 to process the image data according to the agreed-upon processing alternative, and to transfer the image data from the first device to the second device. After transfer, the devices disconnect in step S1012.

20 It should be noted that after transfer of functionality, in either of the first or second embodiments, it is possible for the device that received the functionality to retain it indefinitely. Thus, following receipt of a function, the device can store it in DRAM 16 for use 25 in subsequent image processing jobs, including for use in exporting the functionality to other devices in subsequent image data transfers.

30 Moreover, although it is preferred for the transfer of image data to act as a trigger for the first and second devices to go out and to obtain needed image processing functionality, it is possible for the devices to communicate continuously in the background, over the bi-directional communication link so as to locate and obtain image 35 processing functionality that might be needed in future image data transfers. For example, in a situation where a newer version of some image

processing functionality becomes available at a repository of image processing functionality, it is possible according to the invention for the device to communicate with the repository so as to obtain the updated function, so as to have the updated function available for use in subsequent image data transfers.

It is emphasized that several changes may be applied on the above-described system without departing from the teaching of the invention. It is intended that all the matter contained in the present disclosure, or shown in the accompanying drawings, shall be interpreted as illustrative rather than limiting.

15

卷之三

WHAT IS CLAIMED IS:

5 1. A method for negotiating an exchange
of image processing functionality between first and
second devices over a bi-directional communication
link, comprising the steps of:

10 exchanging device class modules between the
first and second devices, the device class modules
including information concerning relative processing
power of the first and second devices;

15 exchanging function code descriptions
between the first and second devices, the function
code descriptions including information concerning
functionality respectively available in the first
and second devices, together with information
concerning whether such functionality is exportable
to other devices;

20 negotiating an assignment of image
processing functionality between the first and
second devices, with the overall image processing
functionality effecting an efficient image transfer
between the first and second devices; and

25 exchanging image processing functionality
between the first and second devices in a case where
the negotiated image processing functionality
indicates that functionality in one of the first and
second devices is needed by the other of the first
and second devices.

30 2. A method according to Claim 1, further
comprising the step of transferring image data from
the first device to the second device using the
negotiated assignment of functionality, including
functionality that has been exported from one device
to the other.

3. A method according to Claim 1, wherein said step of negotiating an assignment of functionality includes the steps of:

5 determining alternative processing sequences for image data transfer;
applying a cost function to each alternative; and
selecting the alternative with the lowest cost function.

10

4. A method according to Claim 3, wherein said cost function contemplates both image transfer time and image quality.

15

5. A method according to Claim 1, further comprising the step of obtaining function code descriptions for functionality in a repository of image processing functionality, wherein said step of negotiating to assign functionality includes the step of negotiating in respect of the image processing functionality exported from the repository, and wherein said step of transferring functionality includes the step of transferring functionality from the repository.

20

6. A method according to Claim 5, wherein said first and second devices retain transferred functionality for use in connection with subsequent image processing jobs.

25

7. A method according to Claim 1, wherein said first and second devices retain transferred functionality for use in connection with subsequent image processing jobs.

30

8. A network interface card for interfacing between an image processing peripheral

and a local area network, said network interface card including:

5 a network protocol stack for interfacing between the local area network and the network interface card, and for receiving network communications directed to the image processing peripheral;

10 a device-specific application layer that provides device-specific image processing functionality for driving the image processing peripheral, the device-specific application layer receiving network communications directed to the peripheral device from the protocol stack; and

15 a negotiation controller for negotiating an exchange of image processing functionality between a second device on the local area network, the negotiation controller being programmed with process steps for carrying out the method of any one of Claims 1 to 7.

20 9. Computer-executable process steps stored on a computer readable storage medium, the computer executable process steps for negotiating an exchange of image processing functionality between first and second devices over a bi-directional communication link, said computer executable process steps including steps according to any of Claims 1 to 7.

30 10. A storage medium for storing computer executable process steps to effect negotiation of an exchange of image processing functionality between first and second devices over a bi-directional communication link, said process steps including steps according to any of Claims 1 to 7.

ABSTRACT

Negotiation between first and second
5 devices having a bi-directional communication link,
so as to negotiate a low-cost image processing
sequence that effects a transfer of image data from
a first device to the second device. According to
the negotiation, the devices determine whether
functionality present in one device can be executed
10 more efficiently in the other device, so as to
increase image transfer efficiency and maintain good
image quality. If functionality present in one
device is more efficiently executed in the other
device so as to achieve an overall increase in
efficiency in the transfer of image data, then the
15 functionality is transferred over the bi-directional
communication link and the transfer of image data
proceeds by incorporating the transferred
functionality.

20

E-19083

66000000000000000000000000000000

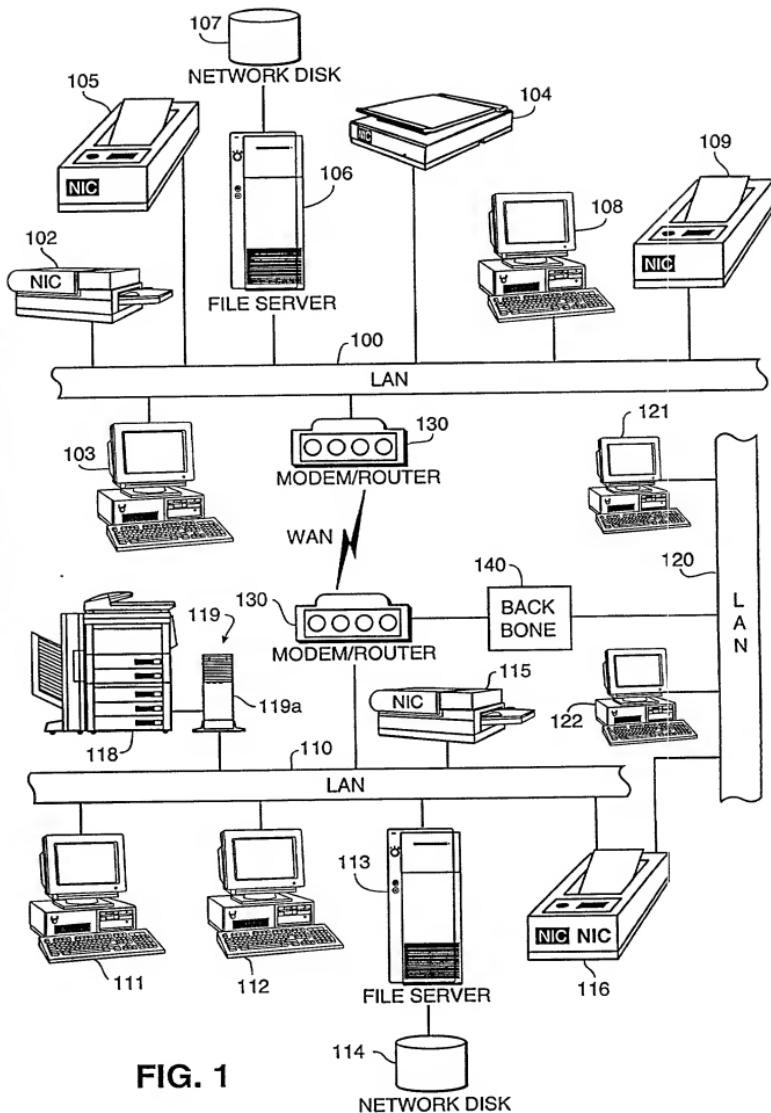


FIG. 1

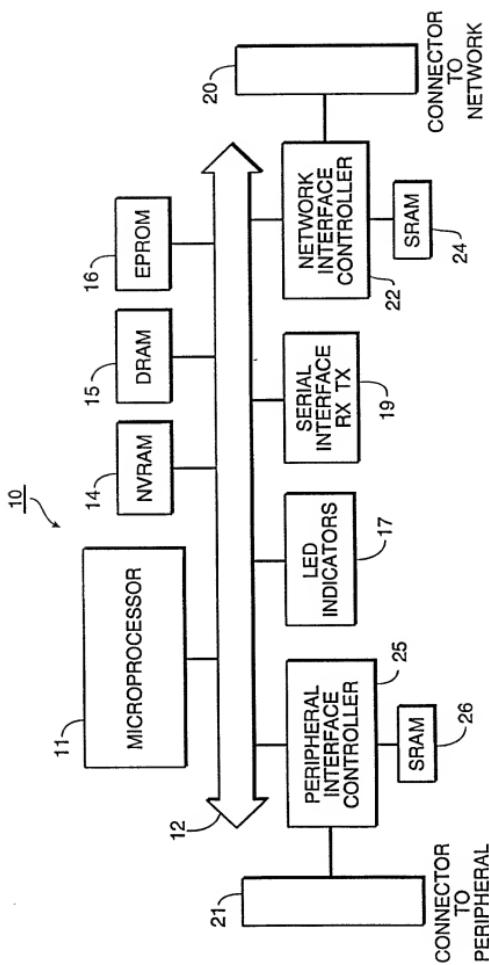


FIG. 2

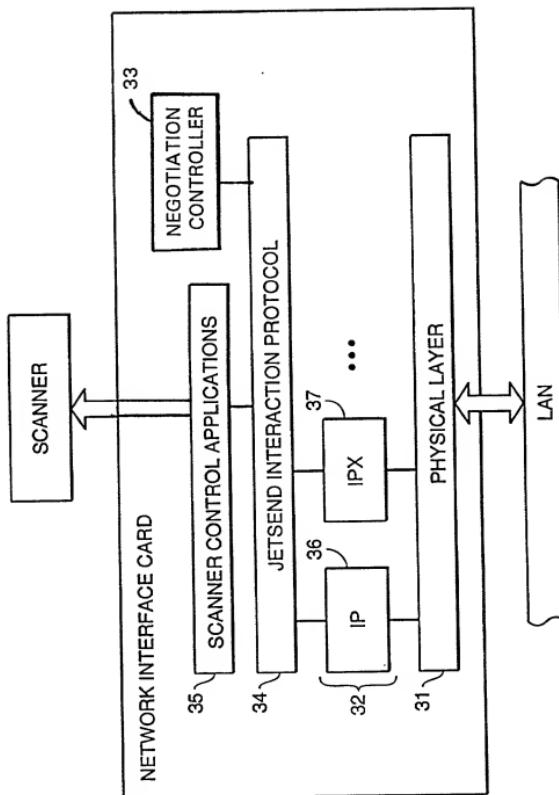


FIG. 3

55555555555555555555

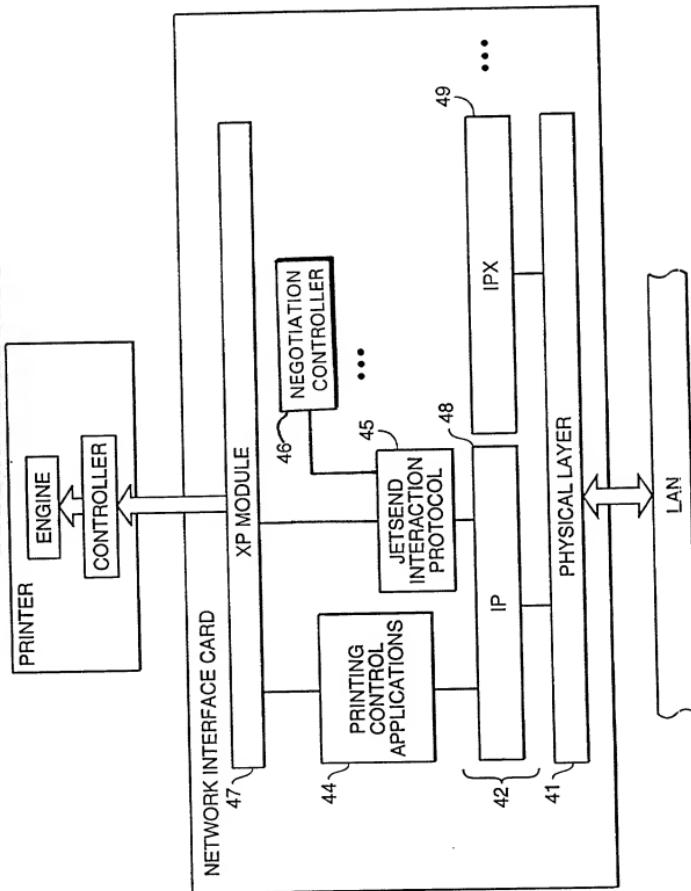


FIG. 4

50

Device Class Module	
Device Type	~51
Device Processing Power	~52
Platform CPU	~54
Platform O.S.	~55
Storage Availability	~56

FIG. 5

60

Function Code Description Module	
Function ID	~61
Version	~62
Name	~64
Purpose	~65
Processing Cost Factor	~66
Size	~67
Exportability	~68

FIG. 6

00000000000000000000000000000000

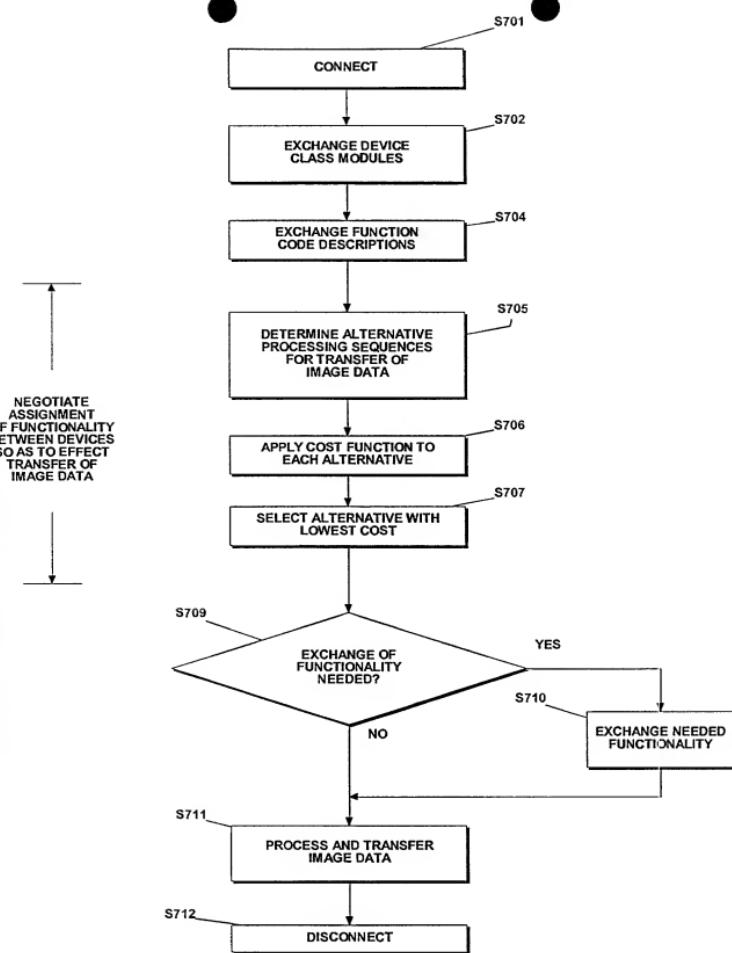


FIG. 7

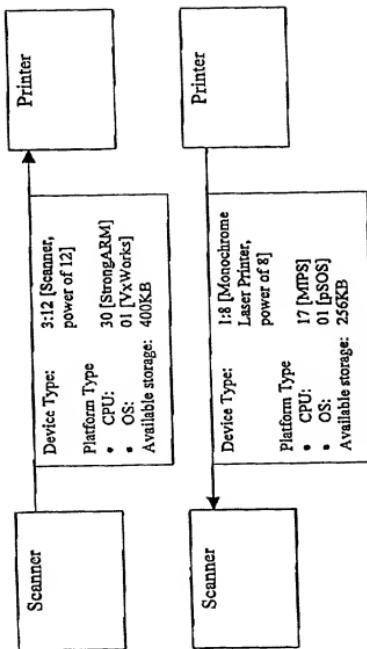


FIG. 8

Scanner

Scanner Function A

- ID: 1:1
[Conversion: Color→Gray]
- Version: 1.00.00
- Name: "Color/Gray conversion"
- Purpose: "Converts 24 bit color images to 8 bit gray scale images."
- Processing Cost: 15 [15 second operation at scanner]
- Size: 64K
- Exportable: Yes: Platform Dependent

Printer

Scanner Function B

- ID: 1:2
[Conversion: Gray→B&W]
- Version: 3.00.00
- Name: "Gray/B&W conversion"
- Purpose: "Converts 8 bit color images to black and white images."
- Processing Cost: 12 [12 second operation at scanner]
- Size: 85K
- Exportable: Yes: Platform Independent

Scanner Function C

- ID: 6:6 [Decompression: JPEG]
- Version: 1.10.09
- Name: "JPEG"
- Purpose: "Decompress images -- best for color photographs. Note that it is a lossy scheme!"
- Processing Cost: 20 [20 second operation at scanner]
- Size: 90K
- Exportable: No

FIG. 9A

Scanner	Printer Function A	Printer
	<ul style="list-style-type: none"> • ID: 1:2 [Conversion: Gray→B&W] 	
	<ul style="list-style-type: none"> • Version: 2.00.23 	
	<ul style="list-style-type: none"> • Name: "Gray/B&W conversion" 	
	<ul style="list-style-type: none"> • Purpose: "Converts 8 bit color images to black and white images." 	
	<ul style="list-style-type: none"> • Processing Cost: 10 [10 second operation at printer] 	
	<ul style="list-style-type: none"> • Size: 40KB 	
	<ul style="list-style-type: none"> • Exportable: No 	
	Printer Function B	
	<ul style="list-style-type: none"> • ID: 5:9 [Compression & Decompression: JPEGLS] 	
	<ul style="list-style-type: none"> • Version: 1.10.09 	
	<ul style="list-style-type: none"> • Name: "JPEG-LS" 	
	<ul style="list-style-type: none"> • Purpose: "Compress images - best for color photographs. Note that it is a loss-less scheme!" 	
	<ul style="list-style-type: none"> • Processing Cost: 25 [25 second operation at printer] 	
	<ul style="list-style-type: none"> • Size: 100KB 	
	<ul style="list-style-type: none"> • Exportable: Yes: Platform Independent 	

FIG. 9B

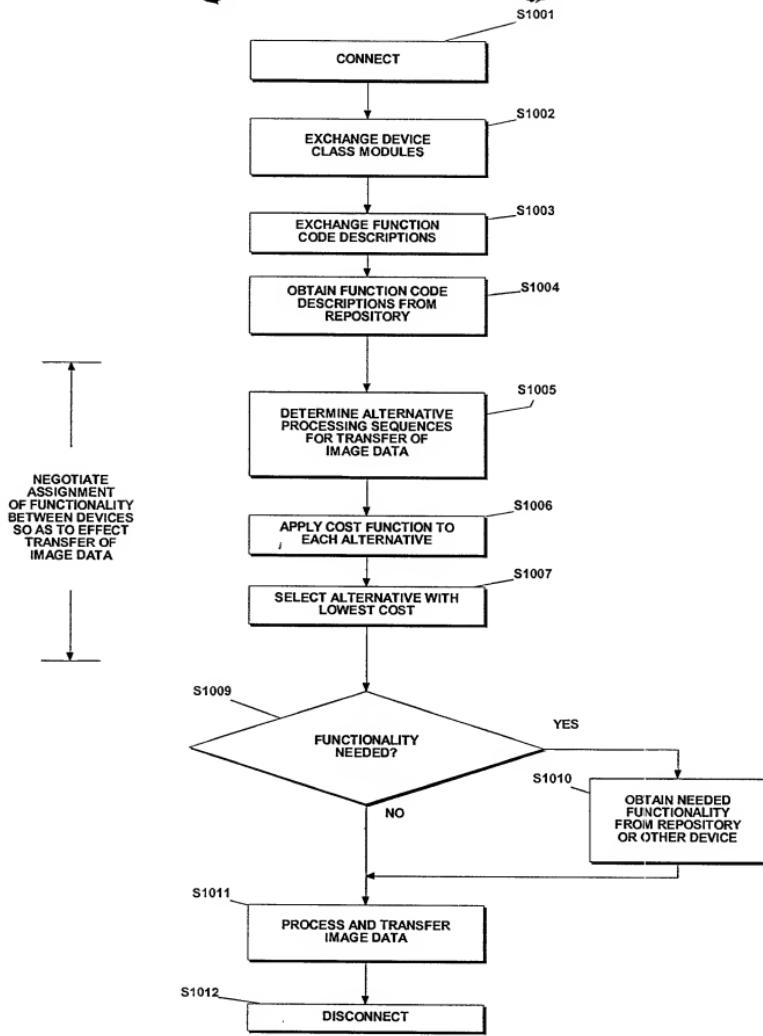


FIG. 10

COMBINED DECLARATION AND POWER OF ATTORNEY
FOR PATENT APPLICATION

(Page 1)

As a below named inventor, I hereby declare that:

My residence, post office address and citizenship are as stated below next to my name;

I believe I am the original, first and sole inventor (if only one name is listed below) or an original, first and joint inventor (if plural names are listed below) of the subject matter which is claimed and for which a patent is sought on the invention entitled NEGOTIATING AN EXCHANGE OF IMAGE PROCESSING FUNCTIONALITY

, the specification of which is attached hereto was filed on _____ as United States Application No. or PCT International Application No. _____ and was amended on _____ (if applicable).

I hereby state that I have reviewed and understand the contents of the above-identified specification, including the claims, as amended by any amendment referred to above.

I acknowledge the duty to disclose information which is material to patentability as defined in 37 CFR §1.56.

I hereby claim foreign priority benefits under 35 U.S.C. §119(a)-(d) or §365(b), of any foreign application(s) for patent or inventor's certificate, or § 365(a) of any PCT international application which designates at least one country other than the United States, listed below and have also identified below any foreign application for patent or inventor's certificate, or PCT international application having a filing date before that of the application on which priority is claimed:

<u>Country</u>	<u>Application No.</u>	<u>Filed (Day/Mo./Yr.)</u>	(Yes/No) <u>Priority Claimed</u>
----------------	------------------------	----------------------------	-------------------------------------

I hereby claim the benefit under 35 U.S.C. § 119(e) of any United States provisional application(s) listed below:

<u>Application No.</u>	<u>Filed (Day/Mo./Yr.)</u>
------------------------	----------------------------

I hereby claim the benefit under 35 U.S.C. § 120 of any United States application(s), or § 365(c) of any PCT international application designating the United States, listed below and, insofar as the subject matter of each of the claims of this application is not disclosed in the prior United States or PCT international application in the manner provided by the first paragraph of 35 U.S.C. § 112, I acknowledge the duty to disclose information which is material to patentability as defined in 37 C.F.R. § 1.56 which became available between the filing date of the prior application and the national or PCT international filing date of this application.

<u>Application No.</u>	<u>Filed (Day/Mo./Yr.)</u>	<u>Status</u> (Patented, Pending, Abandoned)
------------------------	----------------------------	---

**COMPLETED DECLARATION AND POWER OF ATTORNEY
FOR PATENT APPLICATION**
(Page 2)

I hereby appoint the practitioners associated with the firm and Customer Number provided below to prosecute this application and to transact all business in the Patent and Trademark Office connected therewith, and direct that all correspondence be addressed to the address associated with that Customer Number:

FITZPATRICK, CELLA, HARPER & SCINTO
Customer Number: 05514

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

Full Name of Sole or First Inventor Andrew J. Kraslavsky

Inventor's signature John Kraslavsky

Date 4/8/99 Citizen/Subject of U.S.A.

Residence 90 TIMBRE

RANCHO SANTA MARGARITA, CA 92688

Post Office Address c/o Canon Information Systems, Inc.

110 Innovation Drive, Irvine, CA 92612

Full Name of Second Joint Inventor, if any _____

Second Inventor's signature _____

Date _____ Citizen/Subject of _____

Residence _____

Post Office Address _____

Full Name of Third Joint Inventor, if any _____

Third Inventor's signature _____

Date _____ Citizen/Subject of _____

Residence _____

Post Office Address _____

Full Name of Fourth Joint Inventor, if any _____

Fourth Inventor's signature _____

Date _____ Citizen/Subject of _____

Residence _____

Post Office Address _____